July 1990

UILU-ENG-90-2223
DAC-22

(2)

COORDINATED SCIENCE LABORATORY
College of Engineering

CURRENT DENSITY CALCULATION USING RECTILINEAR REGION SPLITTING ALGORITHM FOR VERY LARGE SCALE INTEGRATION METAL MIGRATION ANALYSIS

Hungse Cha

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS None
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UILU-ENG-90-2223 (DAC-22)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Coordinated Science Lab University of Illinois 6b. OFFICE SYMBOL (If applicable) N/A	7a. NAME OF MONITORING ORGANIZATION Rome Air Development Center
6c. ADDRESS (City, State, and ZIP Code) 1101 W. Springfield Ave. Urbana, IL 61801	7b. ADDRESS (City, State, and ZIP Code) United States Air Force AFSC, Rome Air Development Center Griffiss Air Force Base, NY 13441-5700
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Rome Air Development Center 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F30602-88-D-0028
8c. ADDRESS (City, State, and ZIP Code) United States Air Force AFSC, Rome Air Development Center Griffiss Air Force Base, NY 13441-5700	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
Current Density Calculation Using Rectilinear Region Splitting Algorithm for Very Large Scale Integration Metal Migration Analysis

12. PERSONAL AUTHOR(S)
Cha, Hungse

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical	FROM 6/89 TO 6/90	1990 August	61

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Metal Migration, Electromigration, Current Density Calculation, Power Bus Modeling

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

In Very Large Scale Integration (VLSI) chips, metal migration(MM) is an important problem from the reliability standpoint. Furthermore, as the feature size is scaled down, MM becomes an even greater problem because of the higher current densities that would exist in the power and ground busses. Because of the complexity of VLSI power busses, there exists a need for a computer-aided design tool to correctly predict the likely failure site(s). This thesis deals with a primitive splitting algorithm that calculates current density waveforms efficiently. These waveforms are used to find the Median Time to Failure (MTF), a major parameter of concern in predicting MM. This algorithm has been motivated by examining the equipotential plots obtained through finite-element method analysis of simple regions. It has been successfully implemented and tested, and some examples are described.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

CURRENT DENSITY CALCULATION USING RECTILINEAR REGION SPLITTING ALGORITHM FOR VERY LARGE SCALE INTEGRATION METAL MIGRATION ANALYSIS

BY

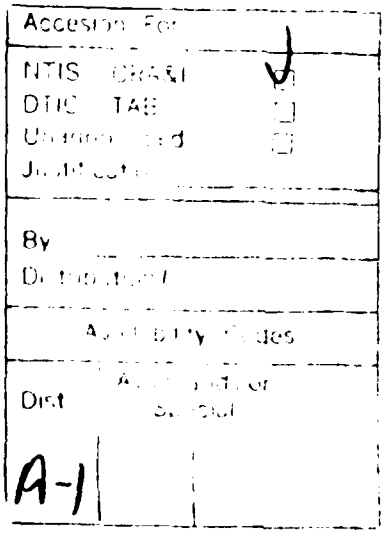
HUNGSE CHA

B.S., California Institute of Technology, 1988

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1990

Urbana, Illinois



ABSTRACT

In Very Large Scale Integration (VLSI) chips, metal migration(MM) is an important problem from the reliability standpoint. Furthermore, as the feature size is scaled down, MM becomes an even greater problem because of the higher current densities that would exist in the power and ground busses. Because of the complexity of VLSI power busses, there exists a need for a computer-aided design tool to correctly predict the likely failure site(s). This thesis deals with a primitive splitting algorithm that calculates current density waveforms efficiently. These waveforms are used to find the Median Time to Failure (MTF), a major parameter of concern in predicting MM. This algorithm has been motivated by examining the equipotential plots obtained through finite-element method analysis of simple regions. It has been successfully implemented and tested, and some examples are described.

ACKNOWLEDGEMENTS

I am extremely grateful to my advisor, Professor Vasant Rao, for his guidance and his insights without which this research would not have been possible. Professor Ibrahim Hajj and Richard Burch provided valuable comments and discussion for which I am grateful. I would also like to thank Russell Iimura and Steven Parkes for their assistance with the existing tools. Finally, I am grateful to the Rome Air Development Center for funding this research.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

In Very Large Scale Integration (VLSI) chips, metal migration(MM) is a great problem from the reliability standpoint. It might cause a chip to fail prematurely, although the devices themselves are fully functional. Furthermore, as the feature size is scaled down, MM becomes an even greater problem because of the higher current densities that would exist in the power and ground busses.

Because VLSI layouts are so complex, there exists a need for computer-aided design (CAD) tools to correctly predict the possible failure site(s) due to MM. The main parameter of concern in MM is median time to failure (MTF), which is a function of current density. This thesis is concerned with the calculation of current density waveforms in metal busses.

Current density in a conductor is linearly related to the gradient of the potential distribution in it. The Laplace equation with mixed boundary conditions may be solved to obtain this distribution. However, since metal busses in VLSI circuits have complex geometry, it is impossible to solve for the distribution analytically. Therefore, a numerical technique such as the finite-element method (FEM) is required. However, an accurate FEM analysis of complex geometry is computationally prohibitive.

Thus, a computationally efficient algorithm is needed to calculate current density in conductors. One such algorithm which has been motivated by examining the

equipotential plots of certain simple shapes called *primitives* has been developed and implemented. Right-angle bends and T-shaped junctions are examples of such primitives. It is found through FEM analysis that the equipotential lines become roughly straight and parallel about one width away from the inner corner. Therefore, forcing the equipotential lines to be straight and parallel at a distance of one width will not cause serious error in current density computations. This allows decomposition of the entire conductor into a set of primitives. FEM analysis may then be performed separately on these regions.

Actually, FEM analysis is performed beforehand to accurately model the primitives. Once primitives in the conductor pattern are identified, a table look-up is used to obtain equivalent resistances and capacitances. Thus, instead of performing FEM analysis, a resistance-capacitance (RC) network may be built which captures all the information about the conductor pattern. This RC network is then solved to obtain the current in the conductor, and this current value, when divided by the width of the conductor, gives the current density.

CHAPTER 2

METAL MIGRATION IN VLSI

Metal migration (MM), also known as electromigration or electrotransport, is a commonly observed phenomenon in light bulbs: the breakage of the filament is due to metal migration. In VLSI circuits, MM can cause the circuit to fail in the following two ways[1]:

(1) at contact windows to the underlying diffusion, etch pit formation can cause failure, and

(2) an open circuit in the power or ground line can cause failure.

This thesis is concerned only with the latter mode of failure.

2.1 Metal Migration Phenomenon

MM is the transportation of metal atoms in a conductor due to the current in it. Metal atoms are thermally activated and become positive metal ions. These ions experience two opposing forces: the electrostatic force of the applied electric field, and the force due to collision with electrons. The force due to the collision is greater than the electrostatic force. Therefore, the metal ions will move in the direction of the electron flow. Eventually, the conductor experiencing MM will fail, and the failure will be manifested as a break in the line.

2.2 Median Time to Failure

Time to failure is found to follow a lognormal distribution. In order to characterize it, the median time to failure (MTF) is used, and it is related to current density as follows [2]:

$$\text{MTF} \propto J_{\text{eff}}\, e^{(Q/kT)} \tag{2.1}$$

where

Q is the activation energy of low-temperature metal self-diffusion along grain boundaries,

K is Boltzmann's constant, and

T is the temperature in degrees kelvin.

The expression for J_{eff} looks very much like the expression for finding the root mean squared value:

$$J_{\text{eff}} = \frac{1}{T}\int_0^T f(J)\, dt. \tag{2.2}$$

The function f is a nonlinear function and is approximated as

$$f(J) = J^{-n}, \tag{2.3}$$

where

$$n = \begin{cases} 1.0 & \text{if } J \leq 10^5 A/cm^2 \\ 1.5 & \text{if } 10^5 < J < 10^6 A/cm^2 \\ 2.0 & \text{if } J \geq 10^6 A/cm^2 \end{cases}$$

The current density value J_{eff} should contain the effects of all possible current densities that might occur. Theoretically, it is possible to compute all possible current loading waveforms at contacts and then use these waveforms to compute all possible current density waveforms. This, however, is computationally prohibitive. Instead, CREST [3] views the currents as stochastic processes and uses probabilistic methods to find expected current waveforms. If these expected waveforms are used, only the expected, not actual, current density waveforms may be found.

It is shown in [4] that MTF due to a random current density depends only on the expected waveform of a nonlinear function of the current density, $E[f(J)]$. Thus,

$$J_{\text{eff}} = \frac{1}{t_0} \int_0^{t_0} E[f(J)]\, dt. \tag{2.4}$$

However, since f is a nonlinear function, the computation of the expected value is difficult. The Taylor series expansion of $E[f(J)]$ is [5]

$$E[f(J)] \approx f(E[J]) + f''(E[J]) \frac{\sigma_J^2}{2}, \tag{2.5}$$

where

$$\sigma_J^2 = E[(J - E[J])^2] \tag{2.6}$$

is the variance. Using $f(E[J])$ instead of $E[f(J)]$ amounts to making a first-order approximation for J_{eff}. Since CREST also outputs variances, the second-order approximation according to Equation (2.5) may be found. Moreover, this approximation is exact in the regions where f may be represented by straight lines. Therefore, using the expected current density waveforms will still lead to an accurate MTF value.

2.3 Metal Migration Scaling

The current trend is toward smaller feature sizes. If technology is scaled by a factor α, the widths of metal lines are scaled by $1/\alpha$, and MTF is scaled by $1/\alpha^6$ [6]. Therefore, MM will become a more severe problem as technology is scaled down, magnifying the need for prediction and compensation for MM.

CHAPTER 3

FINITE-ELEMENT METHOD ANALYSIS

Finite-element method (FEM) analysis of the primitives serves two purposes: it accurately models the resistance of the primitives, and it provides justification for the dimensions of the primitives. In order to model a primitive with an equivalent resistance, the voltage difference and the current between two nodes are needed. In FEM analysis, voltages at boundaries are specified as part of the boundary conditions. Since the conducting boundaries correspond to nodes, the voltage difference is trivially obtained. Computing the current, however, is an involved process because the potential distribution ϕ is needed. Once the potential distribution is found, current density in the conductor can be obtained by using the relations

$$\vec{E}(x,y) = -\nabla\phi(x,y) \tag{3.1}$$

and

$$\vec{J} = \alpha\vec{E}. \tag{3.2}$$

Current is found by summing the current density over a cross section of the conductor. Finally, equivalent resistance is calculated using the equation

$$R_{eq} = V/I. \tag{3.3}$$

The potential distribution which is needed for primitive modeling is also used for primitive specification. Equipotential lines can be drawn using the potential distribution, and upon examining these lines, the distance from the corner of a right-angle

bend or from the edge of contact can be measured, at which point the lines may be assumed to be straight and parallel. This distance is then used to set the dimensions of the primitives.

3.1 The Problem

The electrostatic potential distribution ϕ in any three-dimensional region that is free of sources and sinks satisfies a partial differential equation known as the Laplace equation. For VLSI conductors, the thickness or the depth of the metal is very small compared to the lateral dimensions. Therefore, the variation of ϕ perpendicular to the surface may be ignored, and a two-dimensional Laplace equation may be used instead. For any two-dimensional region, the Laplace equation may be written as

$$\nabla^2\phi(x,y) = 0 \tag{3.4}$$

with mixed boundary conditions

(a) *Dirichlet*: $\phi(x,y) = \varphi(x,y)$ on conducting sides and

(b) *Neumann*: $\frac{\partial\phi}{\partial n} = 0$ on insulating sides

where $\frac{\partial\phi}{\partial n}$ is the normal component of the gradient $\nabla\phi$ along the insulating sides. The Dirichlet boundary condition fixes the conducting sides at certain potentials, and the Neumann boundary condition ensures that no current passes through the insulating sides.

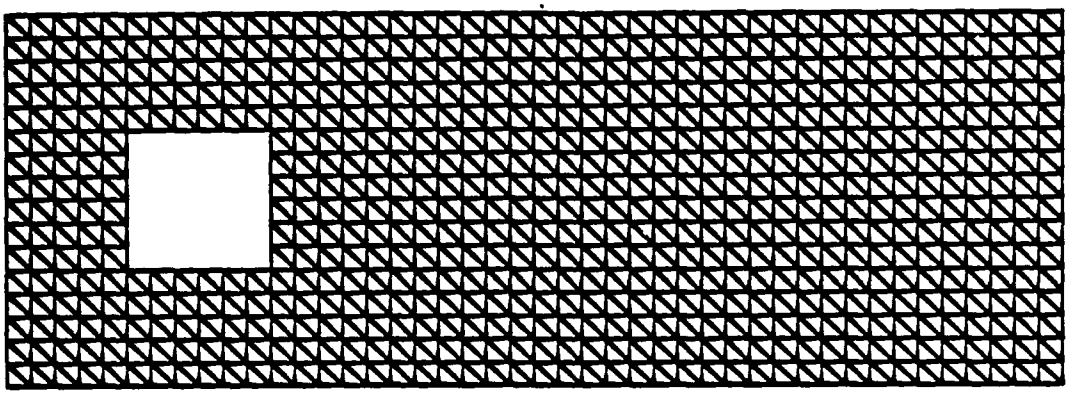


Figure 3.1: Triangularization of a contact region

3.2 The Numerical Solution

The idea behind FEM is to break the region of interest into elements (triangles, rectangles or polygons) which have a fixed number of degrees of freedom, and to use the local, approximate solutions in the elements to build up a solution for the entire domain.

3.2.1 Division into elements

Any polygon or polygons may be used in breaking the region into elements. In addition to the nodes at the vertices, the elements may have additional nodes if more degrees of freedom are desired. For simplicity, only the triangle with three degrees of freedom corresponding to the three vertices will be used as the element in the following discussion.

Figure 3.1 shows a region containing contact divided into triangles. Note that the square corresponding to the contact is not divided because this square is not part of

the metal region. Also note that the elements are right triangles. Using right triangles simplifies matrix formulation, as will be discussed below.

3.2.2 Approximate representation of ϕ inside triangular element

The most general representation of a smooth function over a small area is a polynomial of finite degree. Therefore, ϕ may be represented inside the triangular element as

$$\phi(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy + \alpha_5 x^2 + \cdots. \tag{3.5}$$

Terms are selected on the basis of completeness and compatibility. The potential ϕ should vary linearly along all the sides because ϕ has to match the ϕ of neighboring triangles at two vertices on each side. Since the element has three degrees of freedom, three terms are chosen. Thus, the first three terms of Equation (3.5) are selected to represent ϕ inside the triangular element:

$$\phi(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y. \tag{3.6}$$

Equation (3.6) can be rewritten in terms of the ϕ at the vertices. In other words, ϕ inside the triangular region may be completely determined using the representation for ϕ in Equation (3.6), given the values of ϕ at the vertices. Thus, using generalized coordinates 1, 2, and 3 corresponding to the vertices of the triangular element,

$$\phi(x,y) = f_1(x,y)\phi_1 + f_2(x,y)\phi_2 + f_3(x,y)\phi_3. \tag{3.7}$$

The function f_i is called an interpolation function and has the property that it is

equal to 1 at node i and equal to 0 at the other two nodes.

$$f_i(x,y) = (a_i + b_i x + c_i y)/2\Delta, \quad i = 1,2,3 \tag{3.8}$$

where

$$\begin{aligned} a_1 &= x_2y_3 - x_3y_2, \\ b_1 &= y_2 - y_3, \\ c_1 &= x_3 - x_2, \\ \Delta &= \frac{1}{2}(x_2y_3 - x_3y_2 + x_3y_1 - x_1y_3 + x_1y_2 - x_2y_1), \end{aligned}$$

and the other a_i's, b_i's, and c_i's can be obtained by cyclic permutation of the subscripts 1, 2, and 3. The Δ is simply the area of the triangle.

3.2.3 Matrix formulation over the entire region

It can be shown that of all continuous functions ϕ satisfying the Dirichlet boundary condition, the exact solution of the Laplace equation with both Dirichlet and Neumann boundary conditions is distinguished by the minimum of the potential energy [7]. The potential energy is

$$\Pi(\phi) = \iint_A \frac{1}{2}(\nabla\phi)^2 \, dx \, dy \tag{3.9}$$

where the integration is over the area of the metal region.

In FEM analysis, Equation (3.9) is expressed as a sum of the potential energies in each element

$$\Pi = \sum_{n=1}^{N} \pi_n \tag{3.10}$$

where N is the total number of elements and

$$\pi_n = \iint_{A_n} \frac{1}{2}\nabla^2\phi \, dx \, dy. \tag{3.11}$$

Equation (3.10) holds true if ϕ is continuous over the entire domain. This condition is satisfied with the representation of ϕ given in Equation (3.7).

Differentiating Equation (3.7),

$$\frac{\partial\phi}{\partial x} = \sum_{i=1}^{3} \phi_i \frac{\partial f_i}{\partial x} = (\phi_1 b_1 + \phi_2 b_2 + \phi_3 b_3)/2\Delta, \tag{3.12}$$

$$\frac{\partial\phi}{\partial y} = \sum_{i=1}^{3} \phi_i \frac{\partial f_i}{\partial y} = (\phi_1 c_1 + \phi_2 c_2 + \phi_3 c_3)/2\Delta. \tag{3.13}$$

From Equations (3.11), (3.12), and (3.13), the expression

$$\pi_n = \frac{1}{2}[\phi]_n^T [k]_n [\phi]_n \tag{3.14}$$

can be derived where

$$[\phi]_n = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}_n$$

and the element stiffness matrix is

$$[k]_n = \frac{1}{4\Delta} \begin{bmatrix} b_1^2 + c_1^2 & b_2 b_1 + c_2 c_1 & b_3 b_1 + c_3 c_1 \\ b_2 b_1 + c_2 c_1 & b_2^2 + c_2^2 & b_3 b_2 + c_3 c_2 \\ b_3 b_1 + c_3 c_1 & b_3 b_2 + c_3 c_2 & b_3^2 + c_3^2 \end{bmatrix}.$$

The element stiffness matrix is symmetric and positive definite. Equation (3.10) may be written as

$$\Pi = \frac{1}{2}[\phi]^T [K][\phi] \tag{3.15}$$

where $[\phi]$ is the global potential vector and $[K]$ is the global master stiffness matrix. The element K_{ij} of the global master stiffness matrix represents the i^{th} generalized load due to j^{th} generalized global coordinate.

Since $[K]$ is the summation of symmetric positive definite matrices $[k]_n$, it has these properties also, and Equation(3.15) may be rewritten as

$$\Pi = \frac{1}{2}[\phi_\alpha \varphi_\beta] \begin{bmatrix} K_{\alpha\alpha} & K_{\alpha\beta} \\ K_{\alpha\beta}^T & K_{\beta\beta} \end{bmatrix} \begin{bmatrix} \phi_\alpha \\ \varphi_\beta \end{bmatrix} \tag{3.16}$$

where ϕ_α is the unknown vector and φ_β is the known vector, whose components are given the values as part of the boundary condition on the conducting boundary. Minimizing Π with respect to ϕ_α yields

$$K_{\alpha\alpha}\phi_\alpha = -K_{\alpha\beta}\varphi_\beta. \tag{3.17}$$

The above is solved using LU factorization followed by forward and backward substitutions.

If a right triangle is used as the element, the element stiffness matrix becomes

$$[k]_n = \frac{1}{4\Delta} \begin{bmatrix} c_1^2 & c_2c_1 & 0 \\ c_2c_1 & b_2^2 + c_2^2 & b_3b_2 \\ 0 & b_3b_2 & b_3^2 \end{bmatrix},$$

because b_1 and c_3 are 0. The presence of 0 values in the matrix results in a sparser global stiffness matrix. For a finely discretized region with a very large number of nodes, the solution will require sparse matrix techniques, and the use of a right-triangle element aids in reducing the computation time.

3.2.4 Convergence of FEM

Because FEM is based on the local, approximate solution for the Laplace equation inside the element, the solution obtained is not exact, and some notion of convergence is needed to justify that the solution obtained is valid. The notion of convergence may be approached from two points of view: either the number of degrees of freedom per element can be increased to infinity with the element size fixed, or the size of elements may be allowed to shrink to zero with the number of degrees of freedom per element fixed. Since the former method involves reformulating the local solution, the latter method is generally used.

For the finite-element solution to converge to the exact solution, two criteria must be satisfied [7].

1. The function (in this case Equation (3.9)) must be mathematically defined over the entire domain.

2. The highest derivatives of the dependent variables in the integrand of the function in terms of the generalized coordinates must be able to represent any constant within an element as the element size approaches zero.

The first criterion means that there should not be any discontinuities of local solutions and their derivatives across element boundaries. This is assured because the local representation of ϕ was chosen to be compatible. The highest derivatives are first-order derivatives of ϕ with respect to x and y, and upon examining Equation (3.7),

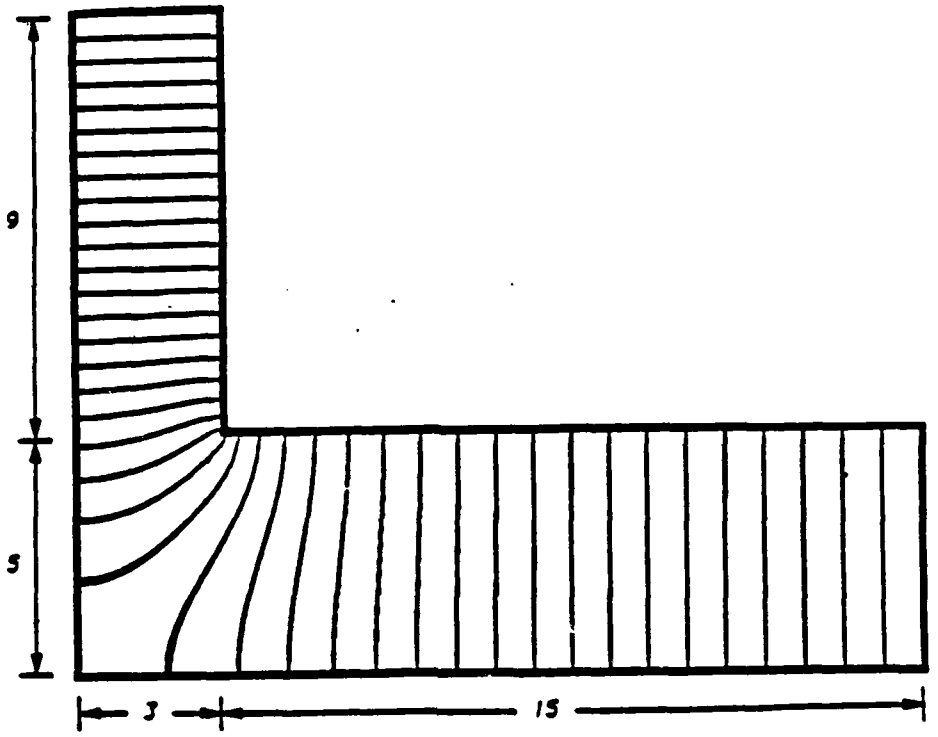


Figure 3.2: Equipotential plot of a region containing a right-angle bend

it is seen that any constant may be represented by these two derivatives, satisfying the second criterion.

3.3 Examples and Primitive Specification

The equipotential lines of an L-shaped region (L), a width change region (W), and contact regions (V) are shown in Figures 3.2, 3.3, and 3.4, respectively. The boundary conditions used are as follows: for the L region, the top side and the right side are set at 1 and 0 V, respectively; for the W region, the left and the right sides are set at 1 and 0 V, respectively; for the V region, the entire via and the right side are set at 1 and 0 V, respectively; the other sides are insulating boundaries. These

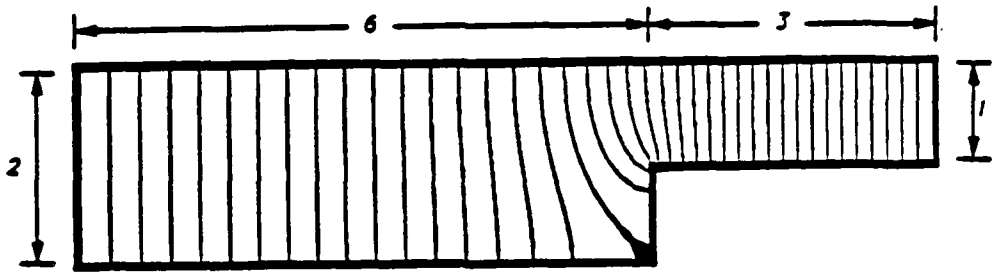


Figure 3.3: Equipotential plot of a region with width change

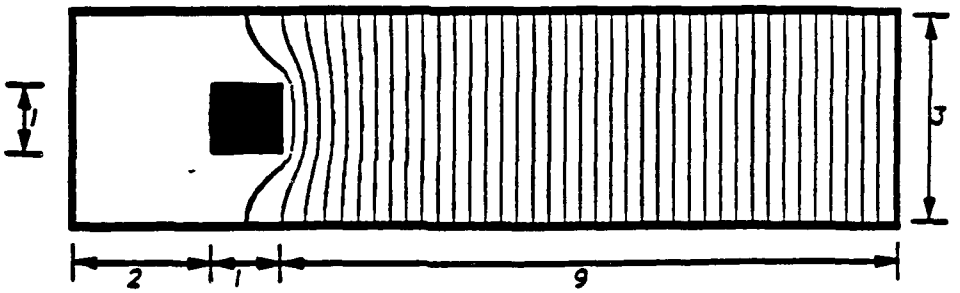


Figure 3.4: Equipotential plot of a contact region

equipotential "lines" have been obtained by mapping vertices within some potential values, e.g., between 0.09 and 0.11 V, to black and the rest to white, so these are actually equipotential "region" plots. It is determined by inspection of the plots that the equipotential lines become roughly straight and parallel about one width away from the inner corner in the L region, one width away from the width change point in the W region, and one width from the edge of via in the V region. This distance of one width defines the primitives, as will be seen in the following chapter.

Although no FEM analysis has been performed in the three-way junction or the four-way junction regions, the same types of straightening out of equipotential lines that occur in the L region were assumed to occur at a distance of one width from the inner corners because these regions can be considered as superpositions of L

type regions. The equivalent circuits were formed based on certain considerations as described in Chapter 4.

CHAPTER 4

PRIMITIVE SPLITTING ALGORITHM

FEM analysis works well for a simple geometry, but for complex conductor patterns, it is computationally unfeasible. An algorithm which overcomes the problems with FEM analysis is based on polygon splitting [8]. The idea is to split the conductor pattern into *primitive* shapes and build an equivalent RC network using RC models of the primitives. Splitting the conductor pattern into a set of primitives is equivalent to forcing the equipotential lines to become parallel at the conducting boundaries between the primitives. Since it was shown in the preceding chapter that the equipotential lines become roughly parallel at a distance of one width away from abrupt changes, the primitive splitting algorithm will incur little error in the actual current density calculations, provided that the primitives are defined properly.

4.1 Primitives

The primitives were developed assuming that the layout conforms to Manhattan geometry, i.e., only right-angle bends are allowed. This assumption simplifies the primitive development and algorithm implementation. Six primitives have been developed and are discussed below.

In the following discussion and in Figures 4.1 – 4.8, R_{sh} is the sheet resistivity of metal conductor, in $\Omega/\square$, and C_{ox} is the capacitance per unit area, in $pF/\mu m^2$. The

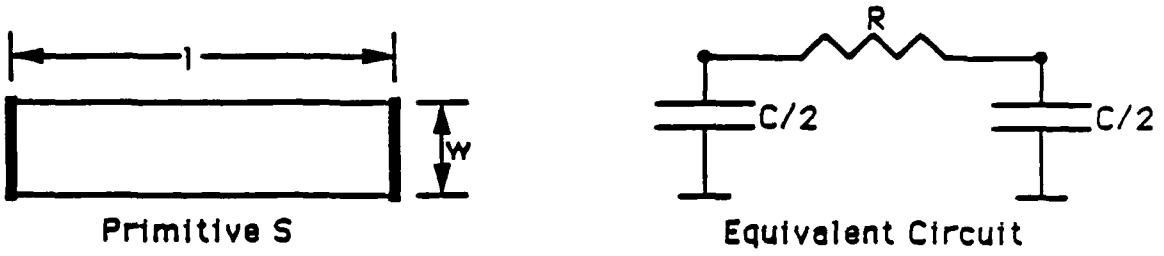


Figure 4.1: S primitive and its equivalent RC model

darkened sides in Figures 4.1 – 4.8 represent the conducting faces of the primitive and the remaining outer sides represent the insulating faces. The equivalent resistances are different for each primitive and are discussed below. The equivalent capacitances, however, are obtained in a straightforward manner. Since only the parallel plate capacitance is considered in the primitives, the capacitance of a piece of metal rectangle is $C = C_{ox}wl$, where w and l are the width and length of the metal rectangle, respectively. If a metal rectangle has only two sides conducting, the capacitance is split between the two nodes corresponding to the two conducting faces. If the metal rectangle has more than two sides conducting, an extra node is created in the middle and the whole capacitance is put there. This extra node is also required for equivalent resistance modeling. The six primitives are defined as follows.

(1) (S) Straight-line segment of length l and width w as shown in Figure 4.1. There are no restrictions on the relative magnitudes of l and w. The equivalent resistance and capacitance are $R = R_{sh}l/w$ and $C = C_{ox}wl$.

(2) (L) L-shaped right-angle bend of smaller width w_1 and larger width w_2 as shown in Figure 4.2. This primitive is defined as having an "inner" rectangle and two abutting *squares*. The equivalent resistance of the entire prim-

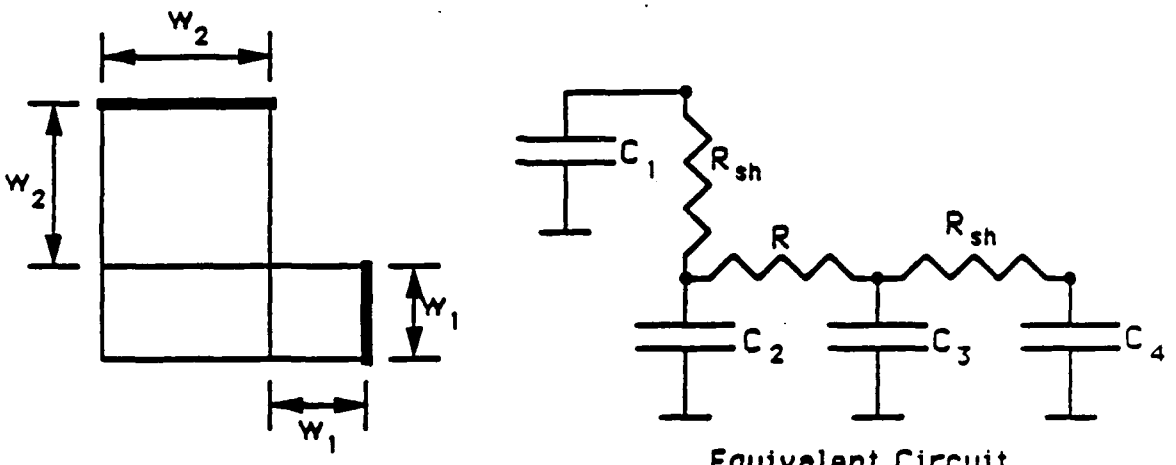


Figure 4.2: L primitive and its equivalent RC model

itive is $R_{eq} = R_{sh}\alpha$, where α is a correction factor as a function of w_1/w_2 due to the bend obtained through FEM analysis (shown in Figure 4.3 [9] by the solid line). This resistance is split into three distinct resistors, two of them being R_{sh}. Therefore, the equivalent resistance of the inner rectangle becomes $R = R_{sh}(\alpha - 2)$. The equivalent capacitances are $C_1 = C_{ox}w_2^2/2$, $C_2 = C_{ox}(w_2^2 + w_1w_2)/2$, $C_3 = C_{ox}(w_1^2 + w_1w_2)/2$, and $C_4 = C_{ox}w_1^2/2$.

(3) (W) Width-change of smaller width w_1 and larger width w_2 as shown in Figure 4.4. This primitive is composed of two abutting squares of different sizes. The equivalent resistance for the entire primitive is $R_{eq} = R_{sh}\beta$, where β is a correction factor as a function of w_1/w_2 due to the width change obtained through FEM analysis (shown in Figure 4.3 by the dashed line). This resistance is split into two distinct resistors, each with resistance $R = R_{sh}\beta/2$. The equivalent capacitances are $C_1 = C_{ox}w_2^2/2$, $C_2 = C_{ox}(w_1^2 + w_2^2)/2$, and $C_3 = C_{ox}w_1^2/2$.

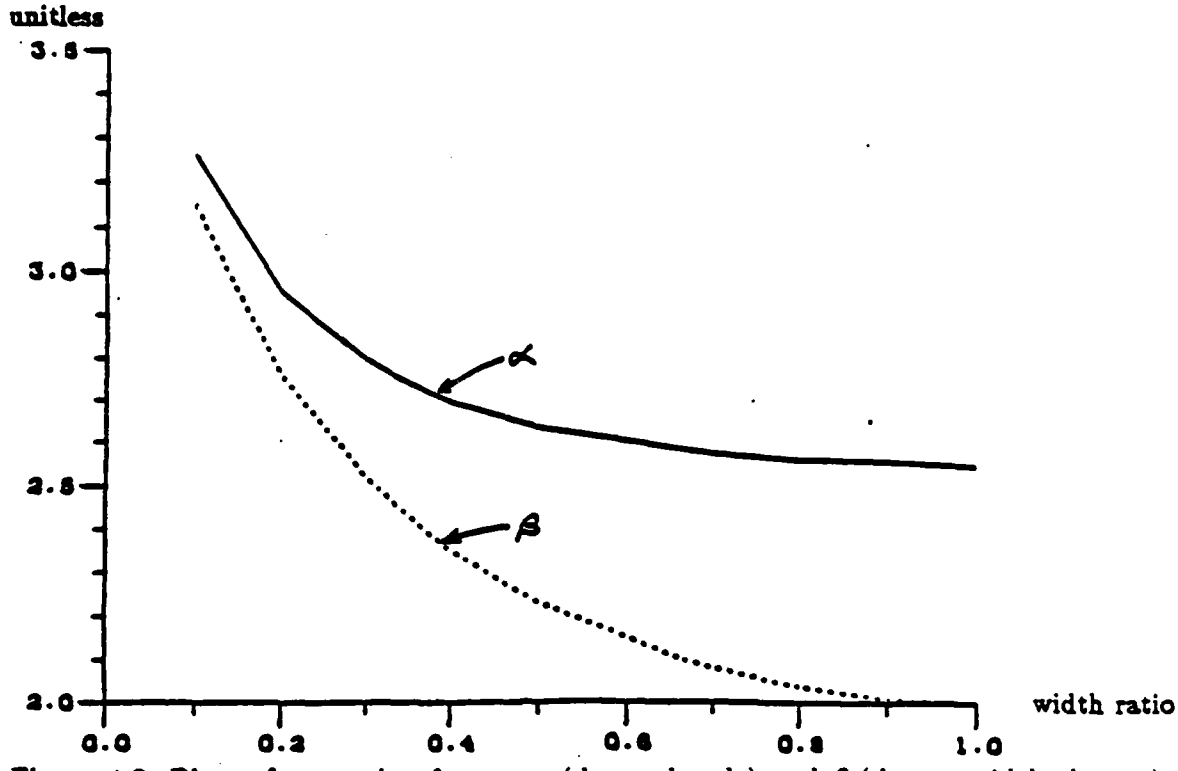


Figure 4.3: Plots of correction factors α (due to bends) and β (due to width change) as functions of width ratio

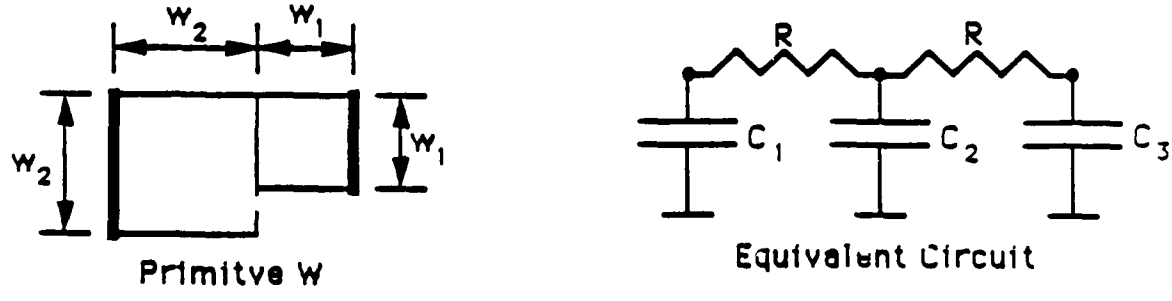


Figure 4.4: W primitive and its equivalent RC model

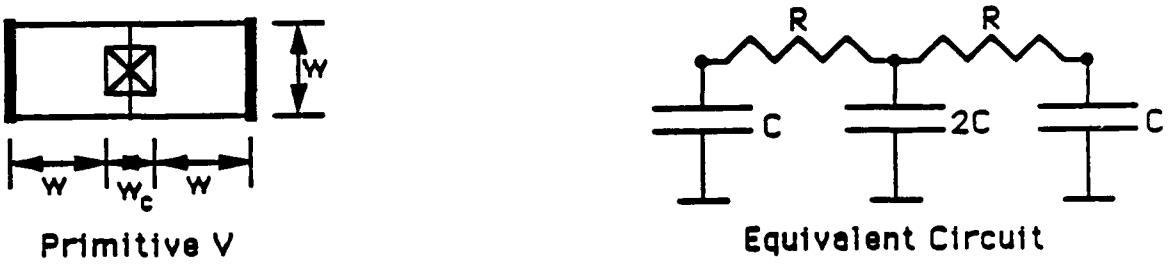


Figure 4.5: V primitive and its equivalent RC model

(4) (V) Via or contact region consisting of two rectangles and a square via of length w_c placed in the center, as shown in Figure 4.5. The length of each rectangle is $w + w_c/2$ where w is the height or the width of the rectangle. The equivalent resistance is $R = R_{sh}\gamma$ where γ is a correction factor as a function of w_c/w due to current crowding at the via obtained through FEM analysis, as shown in Figure 4.6. The equivalent capacitance is $C = C_{ox}[(2w + w_c)w - w_c^2]/4$.

(5) (T) Three-way junction consisting of an inner rectangle and three abutting squares as shown in Figure 4.7. There are no restrictions on the relative magnitudes of w_1 and w_2. The equivalent resistances are $R_1 = R_{sh}\frac{w_2}{2w_1}$ and $R_2 = R_{sh}(\alpha - 2 - \frac{w_2}{2w_1})$. The correction factor α is the same correction factor used for the right-angle bend and is a function of $\min(w_1, w_2)/\max(w_1, w_2)$. The equivalent resistances were derived in the following way. First, R_1's were considered as conducting and R_2 was considered floating. In this case, the resistance is the same as that which would be found in an S primitive. Thus, $2R_1 = R_{sh}w_2/w_1$, from which R_1 is obtained. Then, a combination of R_1 and R_2 is considered conducting while the other R_1 is left floating. In this case,

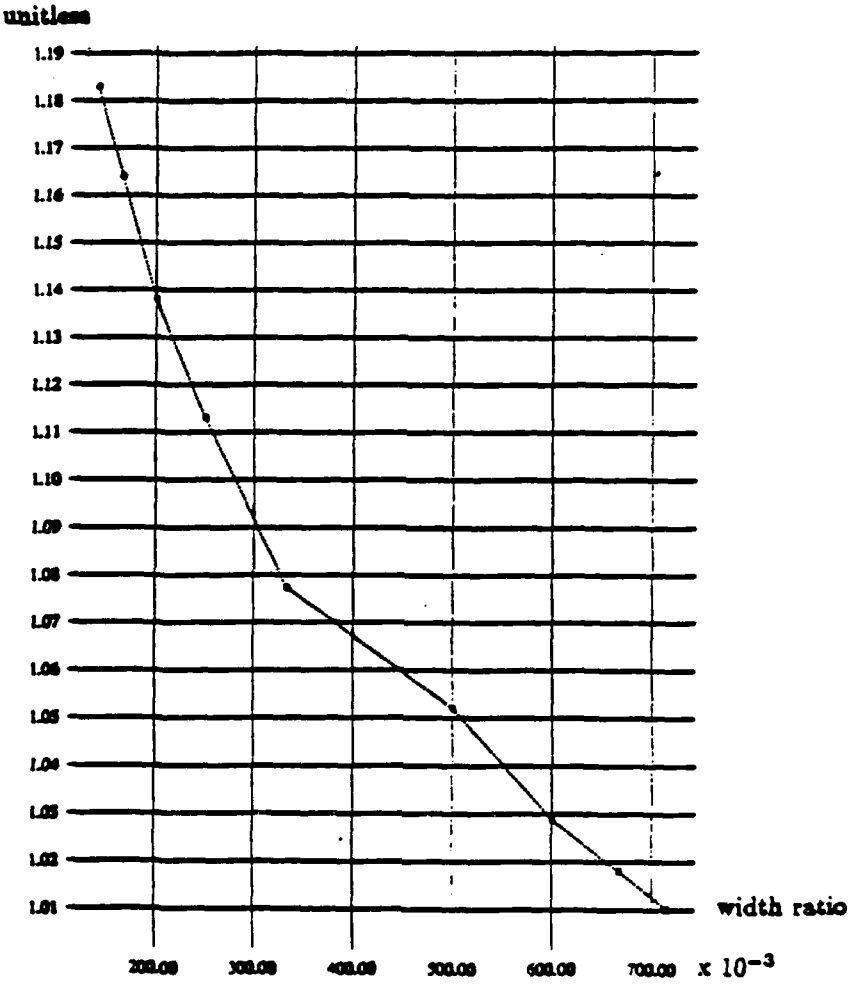


Figure 4.6: Plot of correction factor γ due to current crowding at via

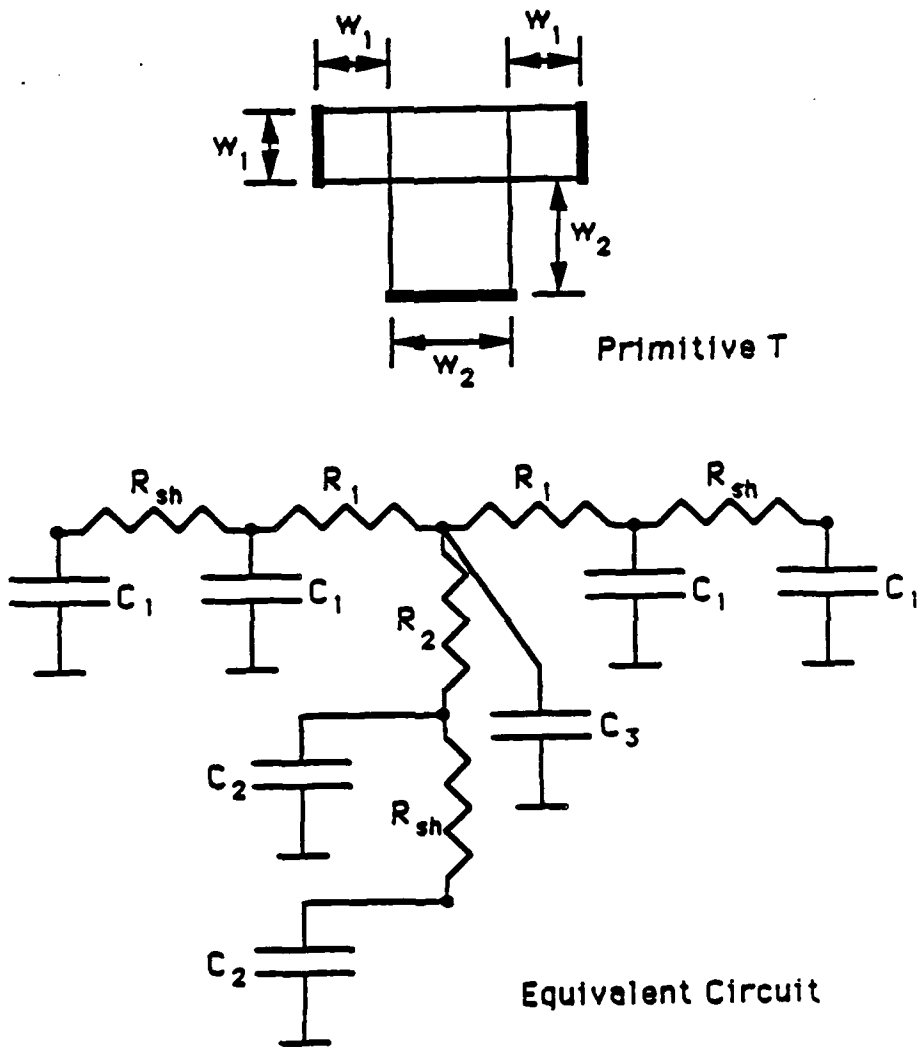


Figure 4.7: T primitive and its equivalent RC model

$R_1 + R_2 = R_{sh}(\alpha - 2)$, and R_2 can be obtained. The equivalent capacitances are $C_1 = C_{ox}w_1^2/2$, $C_2 = C_{ox}w_2^2/2$, and $C_3 = C_{ox}w_1w_2$.

(6) (F) Four-way junction containing an inner rectangle and four abutting squares as shown in Figure 4.8. As in the T primitive, there are no restrictions on the relative magnitudes of w_1 and w_2. The equivalent resistances are $R_1 = R_{sh}\frac{w_2}{2w_1}$ and $R_2 = R_{sh}\frac{w_1}{2w_2}$. First only the straight-line segment consisting of R_1's was considered in order to obtain the equation for R_1. Then, the same procedure was repeated for R_2. The equivalent capacitances are $C_1 = C_{ox}w_1^2/2$. $C_2 = C_{ox}w_2^2/2$, and $C_3 = C_{ox}w_1w_2$.

4.2 The Algorithm

The primitive splitting algorithm is based on the set of primitives defined above. The main procedure is as follows:

```
Procedure Estimate Current Density
begin
    read in OCT cell and create data structure;
    convert geometry to MHS format;
    identify L, T, and F primitives;
    identify W primitive;
    identify V and S primitives;
    create SPICE deck of the geometry;
```

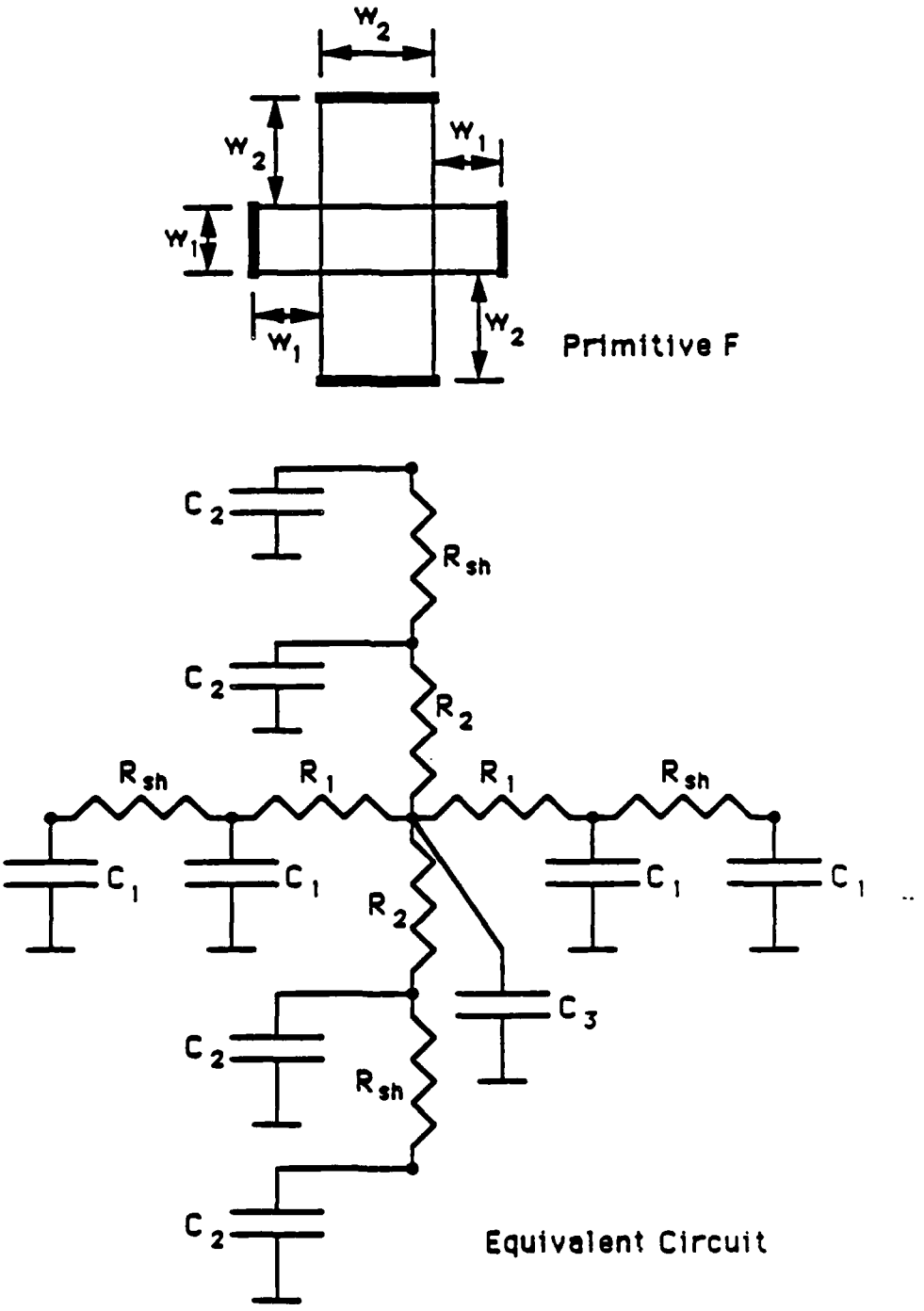


Figure 4.8: F primitive and its equivalent RC model

```
    solve the RC network for voltages;
    calculate current density;
end
```

Each of the major subroutines will be discussed below.

4.2.1 OCT/VEM

The OCT [10] database system and the tools using this system were developed at the University of California, Berkeley. It has its own way of encoding layout information. Therefore, it has routines which support the modification and retrieval of information from the database. VEM is the graphics editor which works in the OCT environment. If metal conductor patterns conform to the set of primitives defined above, there would be no need for VEM. However, in most cases, the conductor patterns contain geometry that cannot be decomposed into a set of valid primitives. In this case, the layout must be modified, and VEM is used for this purpose.

4.2.2 Data structure

The data structure used in the implementation of the polygon-splitting algorithm is a linked list of coordinates, pointers, and associated information.

```
struct RECTANGLE_LIST {
    int LowerLeft_x, LowerLeft_y, UpperRight_x, UpperRight_y;
    int inner_rectangle_flag, primitive_type;
```

```
    int node_num_up, node_num_down, node_num_left, node_num_right;
    int center_node_num;
    struct RECTANGLE_LIST *next, *previous;
    struct RECTANGLE_LIST *right, *left, *up, *down;
    struct RECTANGLE_LIST *contact_link;
}
```

The first four variables are self-explanatory. Inner_rectangle_flag is used to indicate that the rectangle is the inner or the center rectangle in the T, L, or F primitives. Primitive_type specifies one of the six primitives available. The node_num_xxxxx variables hold the node numbers of the corresponding RC network. The center_node_num holds the number of the node that is in the middle of the equivalent RC model of the L, T, or F primitive. The *next and *previous pointers are used to maintain the linked list. The *right, *left, *up, and *down pointers are needed to point to the abutting neighbors in their respective directions. The *contact_link is a link between the contact and the rectangle which encloses it and is used for contact merging and V primitive identification.

There are altogether six linked lists which are maintained in the program that use the above structure: a list for each of the metal one, metal two, poly, and metal one to diffusion contact layers; two lists for metal one to metal two contact layers, one to form V primitives in metal one and the other one to form V primitives in metal two. Although the contact lists do not use any of the flags and any of the pointers except

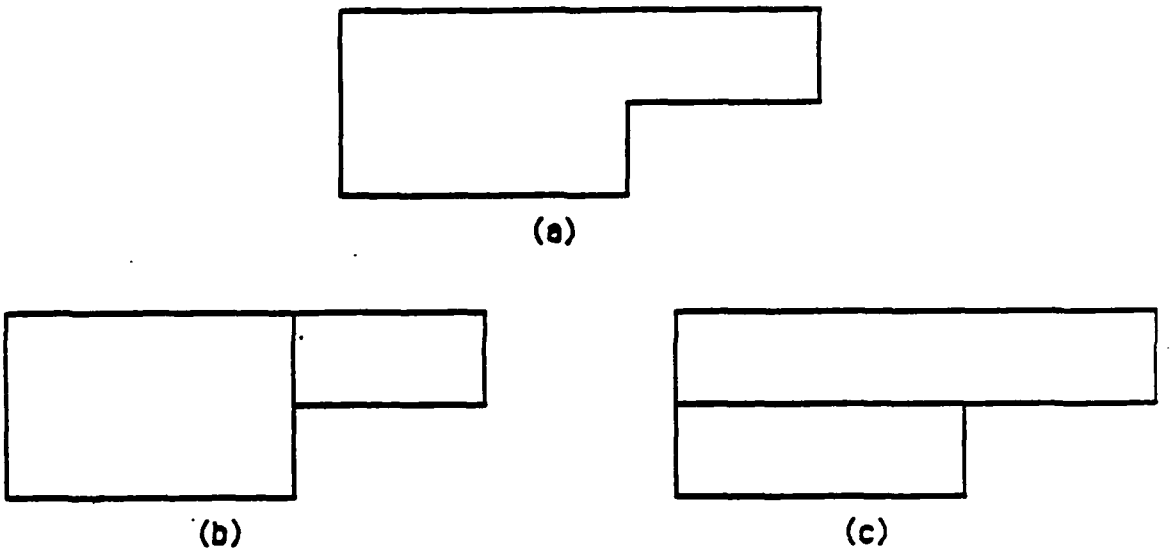


Figure 4.9: (a) a metal region, (b) not in MHS form, (c) in MHS form

for the *contact_link pointer, they use this data structure for simplicity. In addition to the above lists, there is a linked list of labels. This is a straightforward linked list which contains the coordinates and the label.

4.2.3 MHS format

Once the OCT cell is read in and the data structure is built, the rectangles must be put in a maximally horizontal strip (MHS) form. MHS form means that the geometry is described in such a way that it first maximizes the horizontal length and then maximizes the height. For example, Figure 4.9(a) is a polygon which may be described in more than one way. Figure 4.9(b) shows a description not in MHS form, but Figure 4.9(c) shows a description which is in MHS form.

The algorithm which converts an arbitrary description of geometry into MHS form is given below.

```
Procedure MHS
begin
    split current rectangle into left and right pieces if either
      lower left corner x or upper right corner x of another
      rectangle is within the lower left x and upper right x
      of current rectangle
    split current rectangle into upper and lower pieces if either
      lower left y or upper right y of another rectangle is within
      the lower left y and upper right y of current rectangle
    combine rectangles along the x direction to form as large a
      rectangle as possible
    combine rectangles along the y direction to form as large a
      rectangle as possible
end
```

The idea is to break the rectangle under consideration into smaller rectangles whenever a corner of another rectangle is seen within the limits of the rectangle. Then rectangles are combined along the x-direction to maximize the horizontal length before combining along the y-direction to maximize the height.

Figure 4.10: Valid combination of abutting horizontal rectangles

4.2.4 Identification of L, T, and F primitives

The MHS format discussed above is used to simplify the identification of the L, T, and F primitives. The following is the procedure that performs this identification.

```
Procedure Identify L, T, and F Primitives
begin
    identify horizontal and vertical rectangles;
    combine horizontal rectangles to maximize height;
    find inner rectangle of L, T, and F primitives;
    form the complete L, T, and F primitives;
end
```

A rectangle is horizontal (vertical) if its length (height) is at least twice its height (length). Rectangles must be identified as either horizontal or vertical in order to perform the transformation shown in Figure 4.10. The idea is to maximize the width of the conductor the current flowing in it will see. Because the MHS format guarantees that the vertical strips are already as wide as possible, only the vertically abutting horizontal strips need to be checked to see if they can be combined into a wider strip. The requirements for combining are that the rectangles have to be horizontal, that they are abutting vertically as shown in Figure 4.10, and that the resulting rectangles

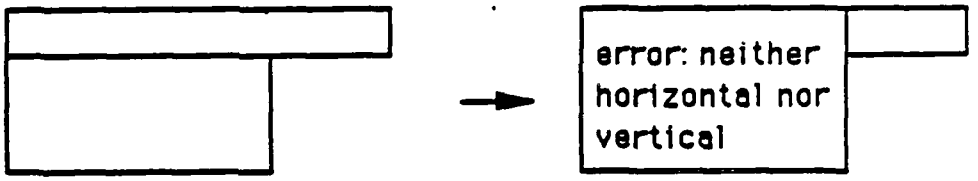


Figure 4.11: Invalid combination of abutting horizontal rectangles

after combination are also both horizontal. Figure 4.11 shows an invalid combination where one of the rectangles ends up being neither horizontal nor vertical. Requiring that the rectangles be either horizontal or vertical does not impose any additional restrictions on the geometry other than that the geometry must be decomposable into a set of primitives because the resulting rectangles must be at least two squares long in order to form valid primitives.

Now, wherever a horizontal rectangle abuts a vertical rectangle, an L, T, or F primitive will be formed. The first step is to form the inner rectangle of these primitives. Only two rectangles, one vertical and one horizontal, are taken at a time and their connectivity is tested. The six valid relative positions of two rectangles and their transformations are shown in Figure 4.12. Each of these cases must be treated separately because the transformed rectangles differ in number and coordinates. In addition to the above six cases, there is a case where a vertical rectangle meets a rectangle already identified as the inner rectangle. In this case, no transformation will be performed. Figure 4.13 shows configurations that are invalid because in these cases proper L, T, or F primitives cannot be formed.

Once the identification of these inner rectangles is complete, the program links the abutting neighbors to each other using *up, *down, *left, and *right pointers.

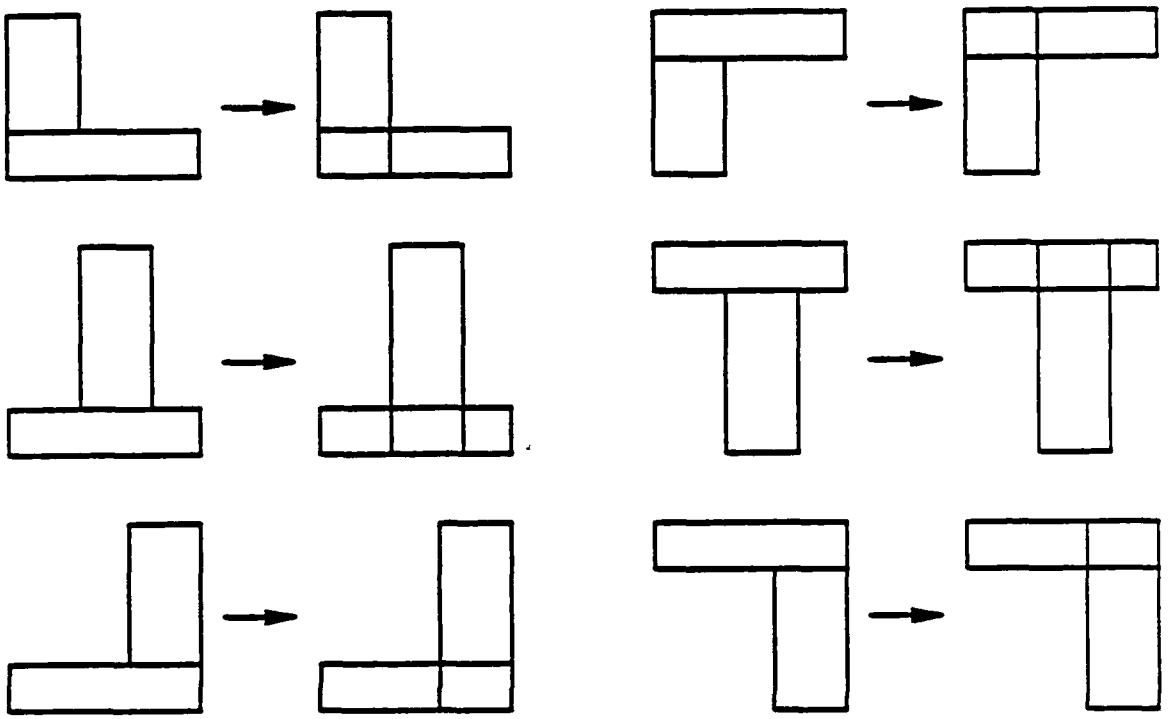


Figure 4.12: Formation of the inner rectangle of L, T, or F primitives

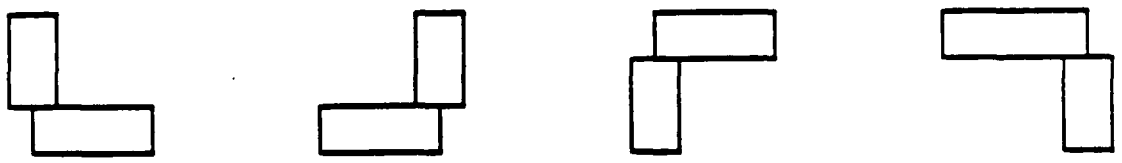


Figure 4.13: Invalid combination of abutting rectangles

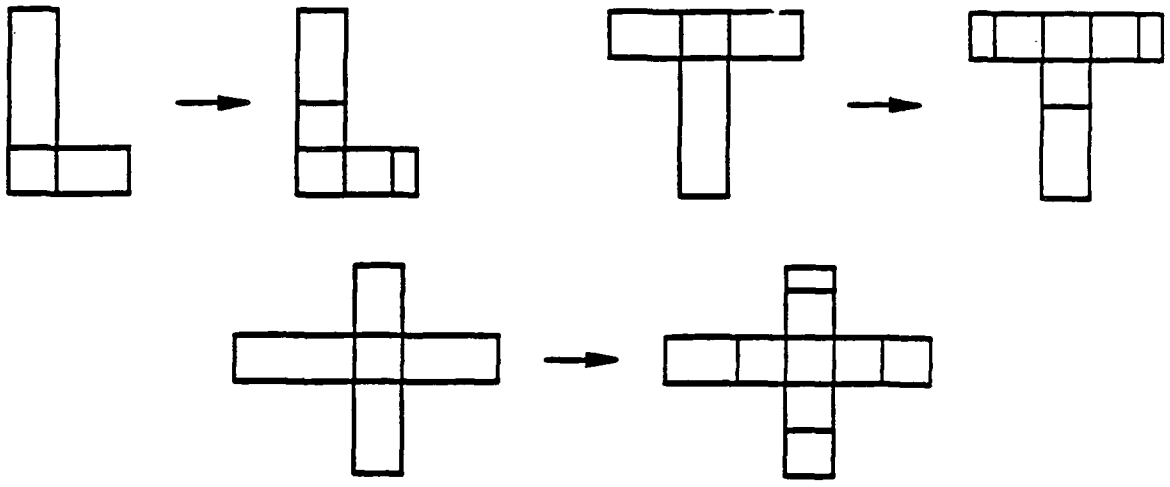


Figure 4.14: Formation of complete L, T, and F primitives

Then, each inner rectangle is visited and the neighbors are split to form a square and a rectangle so that complete L, T, and F primitives may be formed. After all the splittings are done, each inner rectangle is visited again and depending on how many abutting squares there are, it is identified as either L, T, or F primitive. Figure 4.14 shows the transformations to form the L, T, and F primitives.

4.2.5 Identification of W primitive

The identification of the W primitive is not as involved as that of the L, T, and F primitives. Possible locations of W primitives are where two rectangles of different width or of different height abut as shown in Figure 4.15. These rectangles must be straight rectangles which are not part of L, T, or F primitives. Figure 4.15 also shows the transformations that would be performed on these rectangles to create W primitives, given that the rectangles are long enough.

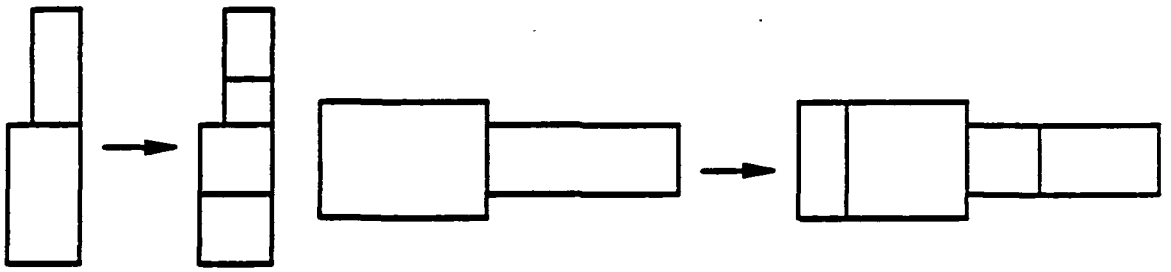


Figure 4.15: Formation of W primitive

4.2.6 Identification of V and S primitives

At this point, all L, T, F, and W primitives have been identified, and the remaining rectangles can only be either V or S primitive. Before the identification and formation of V primitives can proceed, contact merging is performed. This merging is done when two or more contacts are found inside the same rectangle, and the distance between any two of those contacts is less than twice the smaller dimension of the rectangle. The diffusion contacts are merged in a straightforward fashion. The metal one to metal two contacts are merged as well, but one of the metal one to metal two list of contacts is merged using the smaller dimension of metal one as the criterion for merging, and the other list of contacts is merged using the smaller dimension of metal two as the criterion for merging. After merging, the two lists of metal one to metal two contacts are examined to determine the connectivity. The V primitives of metal one will be formed using the contact list where the dimension of metal one was used for merging, and the V primitives of metal two will be formed using the other contact list.

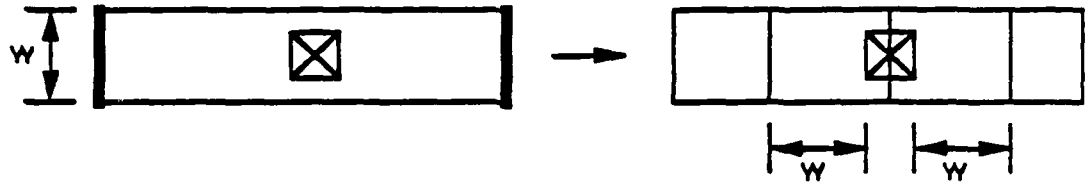


Figure 4.16: Formation of V primitive

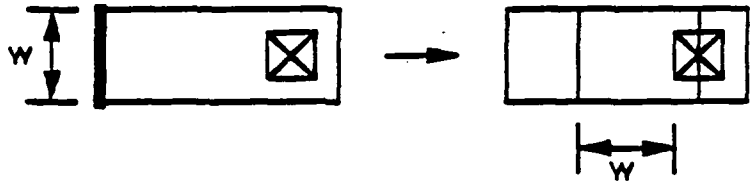


Figure 4.17: Formation of V primitive with relaxed constraint where only one side is conducting

Any straight segment containing a contact or a merged contact is a candidate for a V primitive. Figure 4.16 shows the formation of a V primitive. Normally, there should be enough room to form the complete V primitive. However, in the special case where only one side of the straight segment is conducting, the side that is not conducting may have shorter than the required length as shown in Figure 4.17.

The remaining rectangles that are not part of L, T, F, W, or V primitives are all S primitives.

4.2.7 Creating SPICE deck for simulation

Each rectangle is visited once and the equivalent RC model of the rectangle is written out in SPICE format. For the L, T, and the F primitives, only the inner rectangle has to be treated differently. The abutting square limbs can be treated as S primitives because the equivalent RC models of the limbs are exactly the same as that of the S primitive. This approach is correct because the total resistance from

one end of a primitive to another end is the same, and the total capacitance is the same.

After the RC network has been created, current loading waveforms at contacts created by CREST, a probabilistic simulator, are appended. Because the current loadings at contacts are waveforms and not simple dc values, the current densities will also be waveforms.

4.2.8 Solving the RC network

Although SPICE3 is currently used to solve the RC network, a very efficient algorithm can be used to speed up the process. When an RC network and current loadings are put into matrix form using modified nodal analysis, an $Ax = b$ type of equation results where A is an n x n conductance matrix, x is a vector of unknowns, and b is a vector of sources. This equation can then be solved using Gaussian elimination or Doolittle's or Crout's methods using LU factorization followed by forward and backward substitutions. SPICE assumes that every circuit it is given is nonlinear. Therefore, SPICE reformulates the A matrix, LU factorizes it, and performs forward and backward substitutions at every time step. For a linear RC network, if a fixed time step is used, the A matrix needs to be formulated only once, LU factorization performed only once, and then only forward and backward substitutions need be performed for different b vectors at every time step to obtain the solution vector x. With SPICE, the computation time at each time step is on the order of N^3 where N is the number of nodes. Once the LU factorization is performed, the computation time

at each time point due to forward and backward substitutions is on the order of N^2. Thus a fair amount of savings can be realized over SPICE.

4.2.9 Calculating current density

SPICE3 finds voltage waveforms at specified nodes. In the L, T, F, W, and V primitives, since current density is not uniform in the region, a single value cannot characterize the current density in the region. However, in the S primitive, the current density is uniform, and therefore a meaningful current density value can be obtained. The current density is

$$J = \frac{(V_2 - V_1)}{RW}, \tag{4.1}$$

where R and W are the resistance and the width of the S region, respectively. The current density thus calculated is in A/μm. Note that this is the surface current density. The assumption is that the current density is uniform in the vertical dimension. Therefore, this value would have to be divided by the thickness of the metal to obtain the volume current density, which is required for MTF calculation.

In the L, T, and F primitives, the J_{corn} value, which is the peak current density that exists at the inner corners of the primitives, can be obtained. Analytical calculations [11] as well as FEM analysis show that the current density becomes infinite at the inner corner if it is perfectly sharp. However, in reality, it will not be perfectly sharp but instead will have a small curvature. Given the radius of this curvature r, the width of smaller limb w, the width of larger limb W, and the current density in the

bulk region, J_{bulk},

$$J_{corn} = 1.04 J_{bulk} (\frac{S^2 + 1}{r_n S^2})^{1/3} \tag{4.2}$$

where $r_n = r/w$ is the normalized radius and $S = W/w$ is the width ratio [11]. The radius of curvature at the corners depends on the metal deposition technique and photolithography constraints. The quantity J_{bulk} is the uniform current density found in the narrower limb far enough away from the corner that the equipotential lines are straight and parallel. Scanning Electron Microscopy failure analyses of damage due to MM show that the right-angle bend is not a preferential damage site. However, it might still be useful to know how much larger than the bulk value the magnitude of the current density is at each corner, since local heating may be caused by this singular point.

4.3 Implementation and Examples

The above algorithm has been implemented in the C language on a MicroVax workstation running Ultrix-32 V3.1 (Rev. 9). A user's manual of the implementation can be found in the Appendix. Figure 4.18(a) shows a power bus of a sample cell after it has been decomposed into primitives. Figure 4.18(b) shows a close-up of the box in Figure 4.18(a), and the T, L, S, W, and V primitives can be clearly seen. In the implementation, the V and the S primitives that are not connected to anything, i.e., dangling, are ignored and given no node numbers to save computation time. Vias that do not appear exactly in the middle of the metal bus are handled by treating

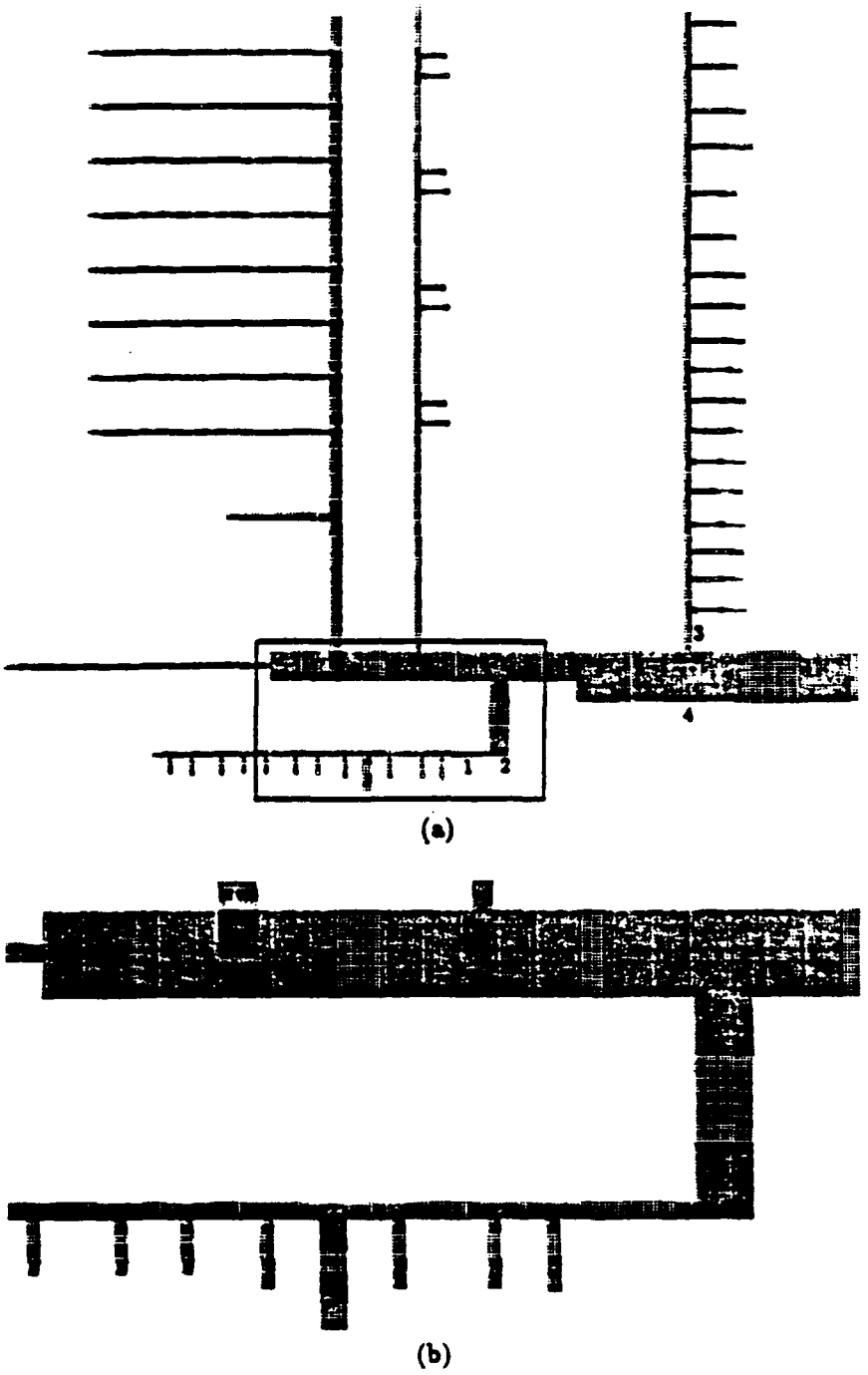


Figure 4.18: Vdd bus of a sample cell after decomposition into primitives, (b) close-up of box in (a)

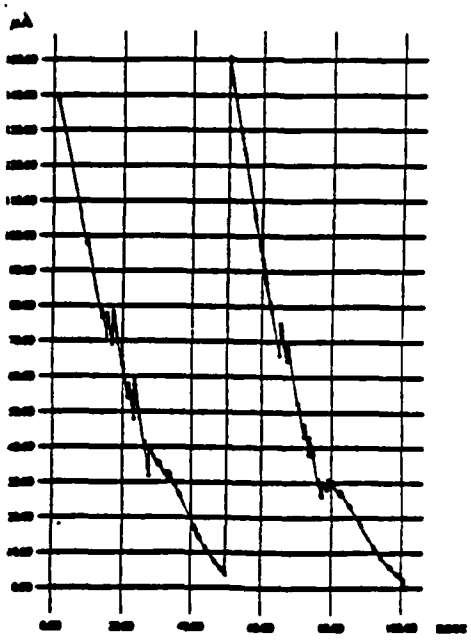


Figure 4.19: Sample current waveform from CREST

them as if they were in the middle to save the modification that would otherwise have to be made.

There are 653 nodes in the equivalent RC network of the bus. It took SPICE3 67 minutes on the SUN 3/50 to simulate the network with the current waveforms provided by CREST. The reason for the long time expended is that, in addition to the inefficient algorithm used in SPICE3 for this type of circuit, the piecewise linear current waveforms (Figure 4.19 shows a sample current waveform) have breakpoints at different times. Breakpoints cause SPICE3 to decrease the step size, resulting in long simulation time.

Figure 4.20 shows four current density waveforms calculated in the numbered regions in Figure 4.18. The numbers match the current density to the region. Current densities J_1 and J_3 are the uniform current densities in straight regions, and current

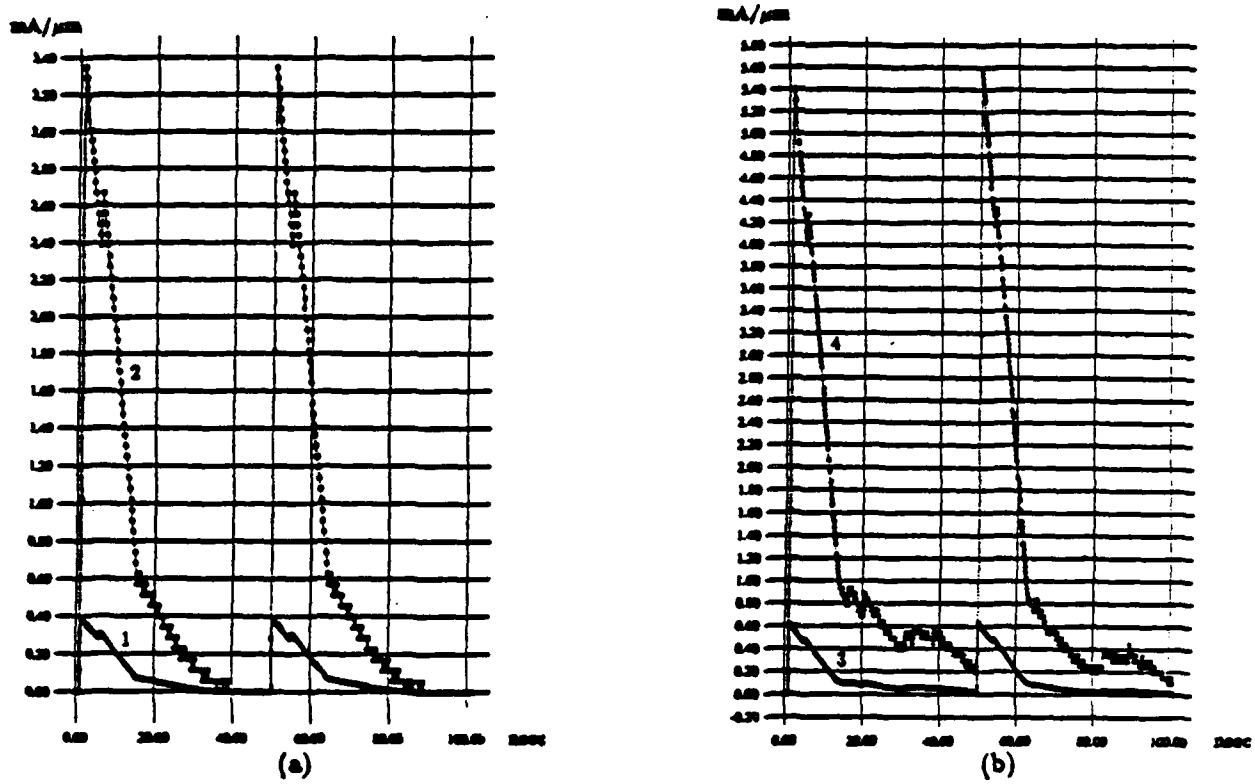


Figure 4.20: Current density waveforms calculated in the numbered regions in the Vdd bus

densities J_{corn2} and J_{corn4} are the current densities at corners. Since J_{corn2} has the same bulk current density as J_1, J_{corn2} should be just a multiple of J_1. The same holds true for J_{corn4} and J_3.

The dimensions of the inner rectangle of the L primitive for which J_{corn2} is found is 5 μm by 16 μm. Thus, $S = 3.2$ and $r_n = 0.002$ for a radius of curvature of 0.01 μm. According to Equation (4.2), J_{corn2} should be 8.51 times J_1. Similarly, the dimensions of the inner rectangle of the T primitive inside which J_{corn4} is found are 41 μm by 6 μm, and J_{corn4} should be 8.83 times J_3.

These predictions are confirmed and can be seen in Figure 4.20. The value of the first peak of J_1 is 0.388 mA/μm and the value of first peak of J_{corn2} is 3.35 mA/μm, so J_{corn2} is 8.62 times J_1. Similarly, the peaks of J_3 and J_{corn4} are 0.614 mA/μm and 5.398 mA/μm, and J_{corn4} is 8.79 times J_3. The reason the values are not exactly the same as predicted is that the J_{bulk} waveform for J_{corn2} is not calculated in the same resistor as the resistor for which J_1 is calculated, but rather in the resistor corresponding to the appropriate limb of the L or T primitive. This still gives the correct waveform since the limb has the same equivalent circuit model as an S primitive. However, since these resistors have very low resistances, the voltage difference between two nodes is very small, and the limited precision of SPICE3 results in slightly different values for currents through the two resistors.

Since CREST provides expected current waveforms at contact points, these current density waveforms are also expected waveforms. As was seen in Chapter 2, expected current density waveforms still give accurate MTF values.

CHAPTER 5

CONCLUSIONS

The primitive splitting algorithm has been successfully implemented and tested. However, the process of editing a given power or ground bus into a geometry conforming to the set of primitives is seen to be a bottleneck. Manual editing is not too time-consuming for a small design, but for a large design, it becomes a very tedious job since VEM is slow to update design changes in a big layout. Although having more primitives will alleviate this problem, some editing will most certainly be necessary for most conductor patterns that would exist in commercial chips. Because there is little understanding of what is and is not a good modification, rules governing the modifications for an artificial intelligence type of program cannot be set down. The designer might wish to use his judgment in modifying some portion of the layout, even if such an artificial intelligence tool is available. As a compromise, a set of macros or remote procedure calls can be written which perform certain types of simple modifications over an area specified by the user.

Another area of concern is the relatively long time it takes for SPICE3 to calculate voltage waveforms. As was discussed in the previous chapter, efficient RC network simulation algorithms will shorten the computation time. In addition, if there were fewer break points in the piecewise linear current waveforms provided by CREST, the computation time could be reduced further. The actual expected current waveform

found by performing exhaustive simulation of the possible current waveforms is seen to be much smoother than the expected current waveforms computed by CREST [4]. Therefore, a low-pass filtering routine can be written to smooth out the jags such as those seen in Figure 4.19.

APPENDIX

JET USER'S MANUAL

Overview

The Current Densitity Estimation Tool (JET) is meant to be used in conjunction with two other tools, iCHARM [12], an extractor, and CREST, a probabilistic simulator. In addition to the above tools, the user must have access to OCT/VEM and should be able to edit in the physical view.

The overall procedure is as follows. The description of the layout in CIF format is run through iCHARM. The extractor identifies power and ground busses and outputs the description in CIF format, which has to be converted into OCT. ICHARM also extracts a circuit description from the layout and outputs the description in SPICE format. This SPICE deck is run through CREST which generates current waveforms at contacts. JET then takes the bus description from iCHARM and the current waveforms from CREST and generates an RC network with current sources in SPICE format. This SPICE deck is run through SPICE3 or any other RC network simulator, and voltage waveforms are generated. These voltage waveforms are then run through a post processor to generate current density waveforms.

iCHARM and CREST

The extractor performs several functions. It identifies the power and ground busses and merges diffusion contacts appearing in the same diffusion region into one contact to which it gives a numerical label. This same label goes into the SPICE description as well as into the CIF file. CREST generates current waveforms at the contacts, and they are identified in JET by the label.

Format

In the CIF description of geometry, diffusion and polycrystalline layers can also be included, but these will be ignored by JET. Only four layers are of importance: CMF (metal one), CMS (metal two), CCA (metal one to diffusion contact), and CVA (metal one to metal two contact). The CCA contacts must have labels if they are to be recognized as current sinks. In addition to CCA labels, there are two special labels "0" and "1," corresponding to the GND power source and the Vdd power source, respectively. These labels must be on CVA contacts, but there is no limit on the number of CVA contacts with these labels on them. The CIF description of the geometry must be converted into OCT format using the tool **ciftooct**.

The CIF description will normally have two subcells, one for GND and one for Vdd. These in turn will become two subdirectories in OCT. Inside the GND cell, only "0" must appear and inside the Vdd cell, only "1" must appear, for obvious reasons.

The CREST output consists of piecewise linear (PWL) current sources in SPICE format. A current source starts with "I" followed by a label to make it unique, usually the label of the contact it is associated with, and then two node numbers, one of them being the label on the contact and the other one being zero. The order of the node numbers will depend on whether it is connected to the Vdd bus or the GND bus. If it is connected to the Vdd bus, the label will appear first. In the PWL description, all the numbers and parentheses must be separated by white spaces.

Primitive Shapes

JET attempts to decompose the layout into S, T, F, L, W, and V primitive shapes which are described below.

S is a straight region in the conductor pattern where two opposite ends of the rectangle are conducting.

L is a right-angle bend. It is composed of a "center" rectangle and two squares connected to adjacent sides of the center piece.

W is a width change. It is composed of two adjoining squares of different sizes.

V is composed of two abutting rectangles of the same width. A contact via appears at the junction between the two rectangles, and each rectangle contains half of the contact. The length of each of the rectangles must be half the contact length plus the width of the rectangle, so that a square may fit exactly

in the region from the edge of the contact to the edge of the rectangle. The restriction on the length of the rectangle is relaxed if it is a dangling piece, i.e., not connected to other metal. In this case, the length of the rectangle may be shorter.

T is a three-way junction. The T primitive is composed of four pieces: the center rectangle and three squares abutting the center rectangle, forming a T shape.

F is a four-way junction, composed of a center rectangle and four adjoining squares.

Since the program can recognize only the above primitives, any geometry that cannot be decomposed into a set of the primitives defined above cannot be processed and the user is informed of this by error messages. Any part of the bus that cannot be processed by the program must be modified manually.

OCT/VEM

The OCT/VEM is used for two reasons: the main reason is the required editing mentioned above; the second reason is that JET obtains geometry directly from the OCT database using OCT functions. The OCT/VEM technology recognized by JET is called "mosis."

Technology File

JET requires a technology file of its own from which it obtains resistance and capacitance information. There should be 5 lines in the file. Each line should contain the variable name and the value, separated by white space. The five variables are as follows:

SIGMA1: sheet resistivity of metal one, given in $\Omega/\square$;

SIGMA2: sheet resistivity of metal two, given in $\Omega/\square$;

ALPHA1: capacitance per unit area of metal one, given in $pF/\mu m^2$;

ALPHA2: capacitance per unit area of metal two, given in $pF/\mu m^2$;

RADIUS: radius of curvature at corner of bend, given in μm.

Error Messages

When an unprocessable geometry is encountered, JET informs the user of the fact. Error messages are of the form "message: coordinates." The coordinates given are in the following order: lower left-hand x, lower left-hand y, upper right-hand x, upper right-hand y.

The following is a list of error messages and their meaning. In the following list, horizontal rectangle means that it is wider than it is tall, and vertical rectangle means that it is taller than it is wide.

neither horizontal nor vertical strip: the rectangle encountered is neither twice as wide as it is tall nor twice as tall as it is wide.

non-horizontal strip after height expansion: after combining two horizontal rectangles abutting vertically to form rectangles that are as tall as possible, the resulting rectangle(s) is(are) not horizontal.

invalid combination during primitive id: improperly formed T, F, or L primitive.

unable to form primitive figure: improperly formed T, F, or L primitives. There is not enough space next to the center piece of the T, F, or L primitives to form squares.

unable to split rectangles due to width change: the rectangles are not long enough to be broken into squares adjacent to the width change boundary.

contact not contained in metal: the contact is not contained in any metal. Sometimes, this might be intentional. This is only a warning, however, and the program will continue execution.

V primitive cannot be formed: the metal region which contains the contact is not long enough to form proper V primitives.

contact not contained in straight region: only straight regions are allowed to contain contacts.

node xxx in CREST output file not found: this should not happen if the same contact labels were given to CREST and to JET.

JET

As was mentioned above, the CIF description from iCHARM must be converted into OCT before JET can be used. Typing

```
ciftooct -f cifinfile CIF_filename
```

will accomplish this task. Once this is done and the current waveforms from CREST are available, JET can be used. To run JET, type

```
jet cell_name CREST_waveform_file.
```

The cell name will normally be either **Vdd** or **GND**. JET will first break the geometry into maximally horizontal strip format, and write over the original cell in this format. The modified geometry will look the same, except that the rectangles will now be in maximally horizontal strip format. JET then tries to identify the primitives. Whenever it encounters an unprocessable shape, it will print out an error message and continue execution until the completion of that phase and then exit. If there are any error messages, the user will have to modify the layout so that JET can properly decompose the layout into primitives. When modifying the power and ground busses, the user should try to keep the width the same and lengthen the metal lines in order to obtain accurate current density waveforms.

When the program has decomposed the busses into primitives, it will write out a cell called `cell_name.p`, where `cell_name` is the name specified when JET was invoked. Different primitives will be mapped to different colors in `cell_name.p`. The T, F, L, W, and V primitives in metal one will be mapped to blue and in metal two will be mapped to purple. The S primitive in either metal is mapped to red. However, if S or V primitives represent part of the geometry that is dangling, they are mapped to green. The program will then ask the user to input a pair of coordinates inside a rectangle of interest. Only S primitives and centers of T, F, and L primitives may be specified. If an S primitive is specified, the program will calculate the uniform current density in the rectangle. If one of the center rectangles is specified, the program will calculate the maximum current density in the center rectangle, which would occur at one of the corners. For the L primitive, there is only one corner of interest. For the T and F primitives, there is more than one corner, and JET calculates the maximum current density for the corner closest to the coordinate specified. This input will result in a .PRINT statement with two voltages in the SPICE deck. The first voltage will be either the left or the top node voltage, and the second will be either the right or the bottom node voltage of the specified piece of metal. When the user is done, he must type `-999 -999` to signal the end of input. When JET completes execution, the SPICE deck will be in a file called `cell_name.spice`.

SPICE3

SPICE3 is used to calculate the voltage waveforms. It was chosen over SPICE2 because it has more precision. To run SPICE3, type

```
spice3 -b cell_name.spice >spice_out_file.
```

The -b flag stands for batch mode.

Post Processor

JET writes another file called `cell_name.tmp_mult`. This file contains the multiplication factors by which the difference of voltage waveforms must be multiplied to obtain the current density waveforms. To obtain the current density waveforms, type

```
postjet cell_name spice_out_file.
```

The cell_name is specified so that the program will know which file contains the multiplication factors. This tool will generate the current density waveforms and put them in files `cell_name.jx`, where `x` is an integer starting from 1. The format of the current density waveforms is in two columns, the first one being the time in seconds and the second one being the current density in A/μm.

REFERENCES

[1] J.R. Black, "Electromigration failure modes in aluminum metallization for semiconductor devices," *Proc. IEEE*, vol. 57, pp. 1587-1594, Sept. 1969.

[2] J. W. McPherson and P. B. Ghate, "A methodology for the calculation of continuous DC EM equivalents from transient current waveforms," Proc. Symp. Electromigration of Metals, in *J. Electrochem. Soc.*, vol. 85-86, pp. 64-73, 1985.

[3] F. Najm, R. Burch, P. Yang, and I. N. Hajj, "CREST – a current estimator for CMOS circuits," *IEEE Int. Conf. Computer-Aided Design (ICCAD-88)*, Santa Clara, CA, pp. 204-207, Nov. 1988.

[4] F. Najm, "Probabilistic simulation for reliability analysis of VLSI circuits," Ph.D. dissertation, Dept. of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign, 1989.

[5] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 2nd ed. New York: McGraw-Hill Book Co., 1984.

[6] D. S. Gardner, J. D. Meindl and K. C. Saraswat, "Interconnection and electromigration scaling theory," *IEEE Trans. Electron Devices*, vol. ED-34, no. 3, pp. 633-643, Mar. 1987.

[7] P. Tong and J. N. Rosettos, *Finite-element Method.* Cambridge, MA: MIT Press, 1967.

[8] M. Horowitz and R. W. Dutton, "Resistance extraction from mask layout data," *IEEE Trans. Computer-Aided Design*, vol. CAD-2, no. 3, pp. 145-150, July 1983.

[9] R. N. Rao, "Resistance extraction and current density computation for Very Large Scale Integration electromigration analysis," M.S. thesis, Dept. of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign, 1988.

[10] Electronics Research Laboratory, University of California, Berkeley, *Octtools Distribution 3.0*, 1989.

[11] F. B. Hagedorn and P. M. Hall, "Right-angle bends in thin strip conductors," *J. Appl. Phys.*, vol. 34, no. 1, pp. 128-133, January 1963.

[12] R. Iimura, "iCHARM: hierarchical MOS circuit extraction with interconnect parasitics," to be submitted as M.S. thesis, Dept. of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign.